What is claimed is:

1) A framework for creating a search system of network resources, comprising:

   a) a first tier defining sources/resources API; and

   b) a second tier providing specific implementations of the sources/resources API.

2) A framework as in Claim 1, wherein the first tier includes a tree of accessible/searchable objects built of NetResource types.

3) A framework as in Claim 2, wherein the tree of accessible/searchable objects built of NetResource types includes at least one virtual NetResource and a plurality of non-virtual NetResources, wherein said at least one virtual NetResource organizes said plurality of non-virtual NetResources into groups and sub-groups.

4) A framework as in Claim 1, wherein the first tier includes manager classes.

5) A framework as in claim 4, wherein the manager classes are selected from the group consisting of download manager, scan manager, and properties manager.

6) A framework as in Claim 1, wherein the first tier includes resource handles.

7) A framework as in Claim 1, wherein the first tier includes threading classes.

8) A framework for creating search systems, comprising an abstraction layer to provide a user with a common interface to heterogeneous network objects, wherein the abstraction layer includes:

    a) source/resource API defined through at least one virtual NetResource class,

5   b) a tree of accessible/searchable objects built of NetResource types,

    c) at least one manager class,

    d) at least one resource handle, and

    e) at least one thread class.

10   9) A search system utilizing an abstraction layer to handle heterogeneous network resources and heterogeneous access methodologies, comprising:

    a) a framework incorporating self-contained modules containing derived NetResource types, and

    b) means for deploying and combining said modules.

15

10) A framework for creating search systems of network access methodologies, comprising:

    a) a first tier defining sources/resources API; and

    b) a second tier providing specific implementations of the sources/resources API.

20

11) A framework as in Claim 10, wherein the first tier includes a tree of accessible/searchable objects built of NetResource types.

12) A framework as in Claim 10, wherein the first tier includes manager classes.

13) A framework as in claim 12, wherein the manager classes are selected from the group consisting of download manager, scan manager, and properties manager.

5

14) A framework as in Claim 10, wherein the first tier includes resource handles.

15) A framework as in Claim 10, wherein the first tier includes threading classes.

10

16) A search system comprising a tree of accessible/searchable objects built of NetResource types, including at least one NetResource subtype.

17) A search system as in Claim 16, wherein said at least one NetResource subtype includes at least one of a virtual NetResource, a resource NetResource, a

15

container NetResource, and a service NetResource.

18) A framework for creating search systems with NetResources organized in a tree structure, comprising:

a) a first Virtual NetResource serving as a root, and

20

b) at least one other virtual net resource, wherein said at least one other virtual net resource branches from said first Virtual NetResource.

19) A method for scanning a plurality of container NetResources, comprising

initiating a call to a common API such that said common API enables a scanning

procedure specific to a particular container NetResource.

5   20) A method for searching container NetResources, comprising initiating a call to a

common API such that said common API enables a searching procedure specific

to a particular container NetResource.